Toiminnallinen määrittely

PROJEKTI

versio 1.0

Versiohistoria

|  |  |  |  |
| --- | --- | --- | --- |
| Versio | Päiväys | Tekijät | Selite (muutokset, korjaukset...) |
| 1.0 | 18.8.2017 |  | Alkuperäinen |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Sisällysluettelo

1. JOHDANTO

1.1 Tarkoitus ja kattavuus

1.2 Tuote

1.3 Määritelmät, termit ja lyhenteet

1.4 Viitteet

1.5 Yleiskatsaus dokumenttiin

2. YLEISKUVAUS

2.1 Ympäristö

2.2 Toiminta

2.3 Käyttäjät

2.4 Yleiset rajoitteet

2.5 Oletukset ja riippuvuudet

3. TIEDOT JA TIETOKANTA

3.1 Tietosisältö

3.1.1 Yksilötyyppi X (kukin omana alakohtanansa)

3.2 Käyttöintensiteetti

3.3 Kapasiteettivaatimukset

4. TOIMINNOT

4.1 Yleistä (tai joku muu sopiva otsikko)

4.2 Järjestelmän toiminnot

4.2.1 Toiminto X (kukin toiminto omaan alakohtaansa)

5. ULKOISET LIITTYMÄT

5.1 Laitteistoliittymät

5.2 Ohjelmistoliittymät

5.3 Tietoliikenneliittymät

6. MUUT OMINAISUUDET

6.1 Suorituskyky ja vasteajat

6.2 Käytettävyys, toipuminen, turvallisuus, suojaukset

6.3 Ylläpidettävyys

6.4 Siirrettävyys/kannettavuus, yhteensopivuus

6.5 Operointi

7. SUUNNITTELURAJOITTEET

7.1 Standardit

7.2 Laitteistorajoitteet

7.3 Ohjelmistorajoitteet

7.4 Muut rajoitteet

8. JATKOKEHITYSAJATUKSIA

# JOHDANTO

## Tarkoitus ja kattavuus

Miksi tämä dokumentti on tehty, kenelle (oman firman suunnitteluporukka vaiko maallikko asiakas ja/tai joku muu taho) tarkoitettu.

Kattaako määrittely koko tuotteen toiminnallisuuden, vai vain jonkin osan siitä. Esimerkiksi käyttöliittymä on jätetty pois tästä dokumentista ja se on kuvattu alustavassa käyttöohjeessa [Viite käyttöohjeeseen].

## Tuote

Rakennettavan tuotteen nimi, tarkoitus ja tavoitteet (hyödyt käyttäjälle).

## Määritelmät, termit ja lyhenteet

Sanat ja käsitteet jotka eivät ole lukijalle (tilaaja/toimittaja) tuttuja tai joiden voidaan ajatella tuottavan sekaannuksia erikoisella käytöllään tai jotka eivät yleisesti ole käytössä tai tiedossa. Nämä kannattaa esittää aakkosjärjestyksessä.

Esim. ASCII-merkistöstä ilmoitetaan onko se 7-bittinen (esim. ISO 10646) tai 8-bittinen (esim. ISO 8859-1).

Tässä voidaan esimerkiksi purkaa englanninkieliset lyhenteet (WEfI = Windowing Environment for Idiots), tai sitten samantien suomentaa nekin (WSfI = Windowing Environment for Idiots, keskivertokäyttäjän käyttöliittymä). Se miten menetellään, on projektikohtainen tyylikysymys.

Taulukko 1.3.1 Dokumentissa käytettävät merkintätavat.

|  |  |
| --- | --- |
| **Lihavointi** | toimintojen nimet  valikon kohdat/nimet  painikkeet |
| *Kursivointi* | käyttäjän syötteet |
| ISOILLA KIRJAIMILLA | tietovarastojen nimet  tiedostojen nimet |
| [hakasuluissa] | viittaukset |

## Viitteet

Järjestelmään tai sen rakentamiseen liittyvät tietolähteet mikäli tarpeen (nimi, versio, päiväys, mistä löydettävissä). Lisätietoja. Tähän liittyviä dokumentteja voivat olla mm. esitutkimus ja vaatimusmäärittely.

Esimerkiksi (viitteissä näkyy tekijä tai aihe ja tekovuosi):

|  |  |
| --- | --- |
| [Dokuty17] | Dokumentoinnin tyyliohje, 10.8.2017, versio 1.0,  Ohjelmistottuotanto, http://www.isict.fi/… |
| [Esitut] | J. Nuutinen, Esitutkimusdokumentti järjestelmälle X, 10.8.2017, versio 1.0. |

## Yleiskatsaus dokumenttiin

Rakenteen kuvaus; mitä missäkin luvussa käsitellään, tärkeä varsinkin mikäli lukija ei ole tottunut lukemaan em. sisällysluettelon mukaisia määrittelyjä. Muutama lause kunkin luvun sisällöstä kertoo paljon enemmän kuin pelkkä sisällysluettelon silmäily. Esimerkiksi seuraavasti:

Dokumentin ensimmäinen luku on johdanto määrittelydokumenttiin. Johdanto kertoo dokumentin tarkoituksen, määriteltävän tuotteen yleiskuvauksen ja käytetyt termit.

Luku 2 kuvaa järjestelmän toiminnan yleisellä tasolla. siihen kuuluvan laitteiston, käyttäjät, järjestelmän riippuvuudet ja rajoitukset.

Luvussa 3 kuvataan järjestelmän tietosisältö eli tietokanta ja tietovirrat.

Luvussa 4 määritellään järjestelmän toiminnot. Kustakin toiminnosta on kuvattu mitä se tarkoittaa, mitä se saa syötteeksensä ja toiminnon suorittamisesta tapahtuvat toiminnot ja/tai vaikutukset.

Luku 5 kertoo järjestelmän ulkoiset liittymät, eli laitteiston, tietoliikenteen ja ohjelmistoliittymät.

Lukuun 6 on kuvattu järjestelmän ei-toiminnalliset ominaisuudet, kuten suorituskyky, vasteajat, käytettävyys ja ylläpidettävyys.

Lukuun 7 on kirjattu suunnitteluun vaikuttavat rajoitteet, kuten standardit sekä ohjelmisto- ja laitteistorajoitteet.

Luku 8 on varattu jatkokehitysajatuksille.

# YLEISKUVAUS

Tämän luvun kuvaukset esitetään yleisellä tasolla ja pyritään niissä mahdollisimman lyhyeen tärkeimmät asiat sisältävään esitystapaan.

## Ympäristö

Laajempi kokonaisuus johon tuote/järjestelmä liittyy, ohjelmisto- tai laitteistoympäristö. Onko tuote itsenäinen vai osa jotakin suurempaa kokonaisuutta.

## Toiminta

Yleinen yhteenveto tuotteen ominaisuuksista (pääkohdat poimittuina 4. luvusta). Yleisesti ohjelman syötteet, toiminta, tulosteet. Tässä ei saa selittää mitään jota ei ole tarkemmin selostettu 4. luvussa.

Mikäli määrittelyssä käytetään tietovirtakaavioita (tai muita hierarkisia kaavioita), liittymäkaavion (korkeimman tason) kaavion paikka on tässä. Ominaisuudet ja toiminnot voi sitten selittää kaavion avulla.

Mikäli ohjelmassa on joitakin erikoisuuksia ne on syytä mainita jo tässä; esim. jollei ole tulostusta kirjoittimelle, jos voidaan käyttää vain hiirellä, jos näyttö on erikoisen kokoinen (taskutietokone).

## Käyttäjät

Käyttäjien (varastomiehet vai myyntipäällikkö vai työnjohtaja vai...) ja käyttöympäristön kuvaus. Ylläpitäjä, onko sellaista? Käyttäjien asema organisaatiossa, koulutus (varsinkin mitä pitää osata, jotta voi käyttää tätä järjestelmää), käyttö (päivittäin vai viikottain vai...).

## Yleiset rajoitteet

Määrittelyä ja suunnittelua koskevat yleiset rajoitteet (lainsäädäntö, sovelluksen kriittisyys, suojaus- ja turvallisuusvaatimukset, liittymät muihin järjestelmiin) koottuina 6. ja 7. luvuista.

## Oletukset ja riippuvuudet

Oletukset, jolloin määrittely on voimassa, esim. tietty käyttöjärjestelmä tai laitteisto (koottu 7. luvusta).

Kiireinen lukija selaa määrittelystä vain 1. ja 2. luvut, sen perusteella hän toteaa kannattaako dokumenttiin perehtyä tarkemmin.

# TIEDOT JA TIETOKANTA

## Tietosisältö

Ohjelman tietosisällöllä on tärkeä vaikutus ohjelman toimintaan. Tämän vuoksi ohjelman sisältämät tiedot ja niiden väliset yhteydet on määriteltävä tarkasti ja täsmällisesti. Eli tarkoituksena on selvittää mitä tietoja järjestelmä käsittelee. Tietokannan tarkka rakenne kuvataan vasta suunnitteluvaiheessa, eikä sitä siten esitetä tässä dokumentissa. Poikkeuksena tästä voi olla hyvin matalan tason järjestelmä tai järjestelmä jonka tiedetään käsittelevän tietoja juuri tietyllä tavalla.

Tietosisällöstä kokonaisuutena kuvataan ensin seuraavanlaisia asioita:

1. tietosisältö korkealla tasolla sekä tietojen väliset yhteydet
2. samaa tietosisältöä (voi siis olla tietokanta) käyttävät muut ohjelmistot tai järjestelmät
3. tukiohjelmisto (esim. varmistukset, toipuminen, testaus)
4. ylläpito-, varmistus- ja suojausnäkökohdat

Tietosisältö ja tietojen väliset yhteydet voidaan kuvata tässä käsitekaavion avulla. Käsitekaavio on ER-, OMT-oliokaavio tms. kaavio, joka kuvaa tietosisällön käsitteellisellä tasolla. Siis ei kuvaa toteutusta, vaan “mallintaa“ reaalimaailmaa. Käsitekaavio myös selitetään yleisellä tasolla sanallisesti tässä yhteydessä (lähinnä suhteet), mutta tarkempi tietosisällön kuvaus on alaluvuissa tietohakemiston muodossa.

Määrittelydokumentissa tiedot on esitettävä sillä tarkkuudella että suunniteluvaiheessa on selvillä vähintään tietojen perusrakenne ja niiden väliset yhteydet. Tavoitteena on todellisuutta (ei toteutusta) täydellisesti (koko sovellusalueen tietosisältö ja piirteet) kuvaava, luettavassa muodossa oleva esitys tietosisällöstä.

Tässä luvussa kuvataan siis jokainen yksilötyyppi (=entity, =tietokokonaisuus, =käsite), yksilötyyppien väliset suhteet sekä yksilötyyppeihin ja suhteisiin liittyvät ominaisuudet (=attribuutit, =yksittäiset “kentät“, =yksittäiset tiedot). Jokainen yksilötyyppi ja sen ominaisuudet kuvataan omassa alaluvussaan kuten kohdassa 3.1.1 on esitetty. Mikäli suhteilla on ominaisuuksia, niille voi tarvittaessa tehdä esimerkiksi oman alaluvun.

Tässä luvussa on myös syytä mainita esimerkiksi jos tiedot pidetään tallessa levyllä, tai vain keskusmuistissa (ei tallenneta ajon jälkeen). Samoin tietohakemistossa käytetyt merkinnät tulee selostaa (satunnaista lukijaa varten); mitä tarkoittavat esimerkiksi (sulkumerkit) ÄÅ äå ö @ = + ...

### Yksilötyyppi X (kukin omana alakohtanansa)

Yksilötyypit kuvataan niiden ominaisuuksien avulla. Jokaisesta ominaisuudesta kerrotaan:

1. tyyppi (kirjain, teksti, desimaaliluku...)
2. koko (pituus, tila jonka se vie. Esim. 8 merkkiä ta jos tiedetään kuvattavan tiettyä tietokantaa, voidaan käyttää sen vakiotyyppejä.)
3. selite (mikä tieto on kyseessä, ei kaikilla pakollinen)

Esimerkki tietohakemistosta.

Lisäksi tarvittaessa kuvataan ominaisuuksien käsittely- ja laskenta­säännöt sekä päivityskriteerit ja -tavat. Esimerkiksi seuraavasti:

1. “Projektin kestoa ei tallenneta erikseen vaan se lasketaan kaavalla Loppupvm - Alkupvm“.
2. “Hakijan tunnus tuotetaan automaattisesti tällä ohjelmalla ja sitä saa päivittää vain tämä ohjelma.“

Tämä tietohakemisto on kuvattava jossakin järkevässä järjestyksessä, esimerkiksi ominaisuuksien sisällön mukaan tai aakkosjärjestyksessä. (Tilanteen mukaan mikä vaikuttaa järkevimmältä, eli miten tiedot löytyvät helpoimmin.)

## Käyttöintensiteetti

Käyttötiheys arvioidaan pahimman tapauksen mukaan. Esimerkiksi “Yhtäaikaisia käyttäjiä on arkisin keskipäivällä enintään 50, muina aikoina keskimäärin 10. Kukin käyttäjä tekee hakuja enintään 10 kpl minuutissa, keskimäärin 3 kpl“. Tuon perusteella järjestelmä mitoitetaan pahimman tapauksen mukaan eli sen on suoriuduttava minuutin sisällä 50 yhtäaikaisen käyttäjän 10 tiedonhausta (vasteajan puitteissa).

Esimerkkejä:

1. Ohjelmaa käytetään päivittäin klo 09-16 välillä eikä lainkaan muulloin.
2. Ohjelmaa käytetään 10 minuutin välein.
3. Ohjelman suoritus käynnistetään klo 23:05 joka päivä.

(Vasteajat ilmoitetaan kohdassa 6.1.)

## Kapasiteettivaatimukset

Kapasiteetti eli tiedonkäsittelykyky. Mm. minkälaisista tietomääristä järjestelmän tulee selvitä. Esimerkiksi: “Järjestelmässä on tallen­nettuna maksimissaan 3000 hakijan tiedot.“

Vaatimuksia, joilla pyritään estämään, että systeemi ei sekoa jos syötteitä tai palvelupyyntöjä tai tapahtumia tulee tiheään tahtiin tai tietoa paljon.

# TOIMINNOT

## Yleistä (tai joku muu sopiva otsikko)

Tässä voidaan mainita kaikille toiminnoille yhteiset asiat, esim. tietyt näppäintoiminnot (Esc, Alt-F4, ^C (CTRL-C), !sh, ^Z, F1...). Eli on otettu kantaa ovatko tuollaiset "vakionäppäimet" käytössä vai eivät. Samoin skandinaavisten merkkien tuki, onko vai ei. Ovatko isot ja pienet kirjaimet samanarvoisia. Voidaanko ohjelmaa käyttää yhtä hyvin hiirellä kuin näppäimistöllä. Tiedostonimien pituus. Älkää käyttäkö sekalanguagea (esim. painonapit ikkunassa "Jatka" ja "Exit").

Yleisesti voi jo tässä alussa ottaa kantaa (tai kullakin kohtaa myöhemmin. pääasia että nekin tulevat mainittua) mm. seuraaviin seikkoihin; ikkunan koon muutos, ikkunan siirto, oletusarvopainonapit, rivinsiirtoko kuittaa, ylipitkän tekstin syöttö tekstikenttään. Sopivassa kohdassa otetaan myös kantaa ohjelman kielisyyteen (dokumentit, koodin kommentit, käyttöliittymä).

## Järjestelmän toiminnot

Ohjelman toiminnot voidaan kuvata esimerkiksi hierarkisesti tietovirtakaavioiden avulla. Näin ohjelma saadaan pilkotuksi riittävän pieniksi palasiksi ja kaikki halutut ominaisuudet tulee käytyä läpi. Mikäli tietovirtakaavioita käytetään, tässä on oikea paikka tason yksi tietovirtakaaviolle (nollataso on liittymäkaavio), jonka avulla ohjelman kuvaus voidaan sitten jakaa sujuvasti alalukuihin. Tietovirtakaavion lisäksi ohjelman toimintoja voidaan kuvata tässä tapahtumalistan (käyttäjän/systeemin) avulla. Myös esimerkiksi näyttökartta (valikkohierarkia, navigointikaavio) on hyvä apu ohjelman toiminnallisuuden hahmottamisessa.

Määrittelyvaiheessa kiinnitetään myös käyttöliittymä, jotta se olisi suunnittelussa ja toteutuksessa selvillä. Käytännössä juuri käyttö­liittymä kuitenkin muuttuu usein suunnittelun ja toteutuksen aikana, kun käyttäjät näkevät “todellisen tilanteen“ paremmin. Niinpä käyttöliittymä voikin olla järkevää kuvata tarkasti jossain muualla kuin tässä dokumentissa (esimerkiksi käyttöohjeessa). Tällöin tämä dokumentti painottuu nimenomaan toimintojen kuvaamiseen ja käyttöliittymästä kuvataan tässä lähinnä toimintojen kuvaamisen kannalta tärkeät osat ja periaatteet.

Mikäli käyttöliittymä kuvataan tässä dokumentissa, niin olennaisia asioita ovat mm. näytöt, ikkunointi, grafiikka, komennot, näppäimistö, raportit. Huomioitavaa näytöllä ovat mm. miten mahdollinen vieritys toimii, miten käyttäjä tietää onko tietoja näytön ulkopuolella (eli onko vieritystarvetta), onko minkäänlaisia opasteita, entä virhe­ilmoi­tukset ja toiminta niiden jälkeen, mahtuvatko kaikki tekstit todellakin näytölle ja kenttiin. Samoin tulostuksen kanssa jos sellainen ohjelmassa on.

Käyttöliittymäkuvien ei tietenkään tarvitse olla millään piirros­ohjelmalla tehtyjä vaan voi ne kuvata erinäköisenä tekstinäkin, koska onhan valikoissa ja graafisissa näytöissä tekstikenttiä. Kuviin kannattaa sijoittaa esimerkkitekstejä mahdollisimman todellisista tilanteista.

Ohjelman toiminnot tulee käydä läpi yksityiskohtaisesti yksi kerrallaan siten että jokainen toiminto esitetään omassa alaluvussaan. Tämä helpottaa viittaamista ja antaa asiakkaalle mahdollisuuden tarkistaa ovatko kaikki vaaditut ominaisuudet määritelty.

### Toiminto X (kukin toiminto omaan alakohtaansa)

Toiminnot voidaan usein kuvata hierarkisesti, jolloin kuvatulla toimintokokonaisuudella on alakohtia. Nämä alakohdat on hyvä kuvata omina alalukuinaan. Esimerkiksi tässä otsikkotasolla kuvattaisiin järjestelmän toimintokokonaisuutta esittämällä tason kaksi tietovirtakaavio ja selittämällä se yleisellä tasolla. Tietovirtakaavion prosessit taas voitaisiin kuvata kukin omassa alaluvussaan.

Toimintoja kuvattaessa mitään "itsestäänselvyyksiä" ei saa unohtaa mainita. Toimintojen kuvaus voidaan kirjoittaa joko jäsennellysti (kuten hakuopas) tai vapaamuotoisena tekstinä (kuten käyttöohje), tilanteen mukaan. Esimerkkejä kannattaa viljellä (voivat olla myös käyttöohjeessa, jolloin niihin viitataan tästä). Yksittäisten toimintojen kuvaus voidaan jäsennellä seuraavasti:

1. toiminnon kuvaus (myös esim. voiko rivinsiirto (enter, return) korvata jonkin muun toiminnon)
2. tarkoitus
3. syötteet (mitä, mistä, paljonko, yksikkö, sallitut arvot)
4. käsittely (tarkistukset, toimintaan vaikuttavat parametrit, käsittelysäännöt)
5. tulosteet sekä
6. virhetilanteet (miten toimitaan, miten ilmoitetaan käyttäjälle, mitä tehdään virhetilanteen jälkeen).

# ULKOISET LIITTYMÄT

## Laitteistoliittymät

Käyttääkö järjestelmä ulkoisia laitteistoja, esim. tulostin. Jollei tulostustoimintoa ole on sekin oleellista mainita.

## Ohjelmistoliittymät

Käyttääkö järjestelmä tai liittyykö se muihin ohjelmiin/ohjelmistoihin (esim. ulkopuoliset tietovarastot). Esimerkiksi jos järjestelmä on osa oinfo-järjestelmää, tässä tulee mainita missä rajapinta fyysisesti on (esim. Ingres-tietokanta nimeltä abc) ja mistä sen määritykset löytyvät (esim. LAKEsta Ingres ohjekirja nimeltä xyz).

Mikäli liittymä on suoraan johonkin ohjelmaan, sen tarkka versionumero tulee merkitä tähän. (Ei päällekkäisyyttä kohdan 7.3 kanssa.)

## Tietoliikenneliittymät

Käyttääkö järjestelmä tietoliikenneyhteyksiä, esim. modeemi tai lähiverkko (ja mitä tyyppiä). Mikä hoitaa tietoliikenteen, teidän sovelluksenneko (toiminnot-lukuun) vai jokin muu, esim. käyttöjärjestelmä.

# MUUT OMINAISUUDET

## Suorituskyky ja vasteajat

Staattiset: esim. montako päätettä, montako tiedostoa. Dynaamiset: esim. montako tapahtumaa aikayksikössä.

Vasteaika (saantiaika) kerrotaan, vaikkapa tyyliin "95 %:ssa alle 1 s., enintään 5 s.". Toki moniajokäyttöjärjestelmissä on hankalaa ennustaa vasteaikoja; kuormitusta voi mitata vaikkapa SunOS:n komennolla "uptime" tai "top".

Vasteaikavaatimukset voivat jossakin reaaliaikajärjestelmässä olla myös sellaisia, ettei määriteltyjä lyhempiä aikoja saa esiintyä. Esimerkiksi lyhin sallittu vasteaika on 0,2 sekuntia ja pisin sallittu 20 sekuntia.

Mikäli on olemassa sopiva vertailukohta, esimerkiksi benchmark-testi, niin se mainitaan vertailuarvoineen. Joka tapauksessa jokin suure jota vastaan voidaan suorituskykyä ja/tai vasteaikoja mitata.

(Vasteajat mieluummin tähän kuin 3. lukuun.)

## Käytettävyys, toipuminen, turvallisuus, suojaukset

Käytettävyys on esim. matkapuhelinkeskuksilla: suurin sallittu käytöstä poissaoloaika vuodessa on kolme (3) minuuttia.

Toipuminen, elpyminen on etenkin tiedonhallintajärjestelmiä käytettäessä olennaista. Miten on hoidettu esim. levyrikkojen ja sähkökatkosten aiheuttamat vaaratilanteet tiedoille.

Turvallisuus tarkoittaa etupäässä yhteistyökykyisten ja -haluisten henkilöiden inhimillisiä vahinkoja (esim. "väärien" tiedostojen tuhoaminen, vaikkapa päälle kirjoittamalla), vasta toissijaisesti ilkeämielisiä laillisen yhteiskuntajärjestelmän vastaisesti toimivia yksilöitä.

Voidaan ottaa kantaa esimerkiksi tietoliikenneyhteyden suojauksiin; tarvitseeko palvelimen ja asiakkaan välillä liikkuva tieto salata.

Suojaukset tiedostoille tai niiden osille (rahaliikennetiedot, salasanat, hetu,...). Käyttöoikeudet, salakirjoitus, loki.

Yleensä; onko noihin seikkoihin otettu kantaa vai ei.

## Ylläpidettävyys

Tärkeä kohta ellei erillistä ylläpito-ohjetta tehdä. Ylläpito voi olla korjaavaa tai lisäävää. Kohta tehdään etenkin mikäli järjestelmällä on erillinen ylläpitäjä. Esim. mitkä kohdat/ominaisuudet ovat jälkeenpäin helposti muutettavissa (käyttöliittymä, tietokanta, tietoliikenne­protokolla,...) ja mitä tulee ottaa huomioon.

## Siirrettävyys/kannettavuus, yhteensopivuus

Onko otettu millään tavalla huomioon.

Mihin muihin järjestelmiin sovellus sopii (esim. yhteensopivuus käyttöjärjestelmien tai ikkunointiympäristöjen suhteen).

## Operointi

Tarvitseeko käyttäjän tehdä erityisiä muita toimenpiteitä kuin käyttää järjestelmää, ellei erillistä ylläpitäjää ole. Esim. poistaa vanhoja loki- tai väliaikaistiedostoja tai tehdä muita "siivouksia" (core dumped). Tai asettaa hakupolkujaan tai ympäristömuuttujiaan.

# SUUNNITTELURAJOITTEET

## Standardit

Mitä standardeja (suosituksia, ohjeita, säädöksiä, direktiivejä) liittyy toteutettavaan järjestelmään (esim. eri dokumentit, ohjelmointikieli). Esim. ANSI/IEEE 1016-1987.

Käytännössä ko. lähteet mainitaan tässä, ja niissä on viite kohtaan 1.4. Esimerkiksi käytettävä protokolla on suosituksen RFC12345 mukainen [RFC96] (kyseinen suositus on vuodelta 1996).

## Laitteistorajoitteet

Esim. nykyinen laitteisto, kun uutta ei haluta hankkia. Selitetään ominaisuudet. Esim. prosessorina Intel …, 4 Gt RAM, 33 Gt kiintolevy (jossa vapaata levytilaa sovellusta varten xy Mt, tietokantaa varten yz Mt, ja vapaata työtilaa zx Mt).

Nykypäivänä laitteistovaatimuksiksi riittää joissakin tapauksissa että esim. Windows 10 pyörii koneessa, tällöin ei tarvitse mainita muita teknisiä tietoja.

Laitteistosta voidaan ilmoittaa minimikokoonpano sekä suositeltava kokoonpano.

Myös mahdollisen testausympäristön luomisen takia nämä tulee ilmoittaa yksityiskohtaisesti.

## Ohjelmistorajoitteet

Esim. nykyinen ohjelmistoympäristö, kun uutta ei haluta hankkia. Selitetään ohjelmat. Esim1. tietokantana … Esim2. käyttöjärjestelmänä …

Tässä kohtaa siis puhutaan esim. Windows 10:sta ja muualla dokumentissa vain yleisesti Windowsista (ei versiopäivitysongelmia moneen paikkaan).

Myös mahdollisen testausympäristön luomisen takia nämä tulee ilmoittaa yksityiskohtaisesti.

(Ei päällekkäisyyttä kohdan 5.2 kanssa.)

## Muut rajoitteet

Muita mahdollisia rajoituksia (yleensä käyttäjän/tilaajan taholta).

# JATKOKEHITYSAJATUKSIA

Matkan varrella mieleen tulleita ajatuksia, joita ei tämän projektin puitteissa kuitenkaan määritellä tarkemmin tai toteuteta. Esimerkiksi ajan puutteen, rahan puutteen, resurssien puutteen tai taitojen ja osaamisen puutteen takia.